BITS F464 Machine Learning

Assignment 1

Date : 15th March, 2020

Team Members

Durba Satpathi	2019A7PS0972H
R Adarsh	2019A7PS0230H
Navdeep Singh Narsinghia	2017B5A71675H

Problem 1C Linear Perceptron

3.1. Model Description and implementation

The Perceptron is a linear machine learning algorithm for binary classification tasks. The perceptron algorithm can be used to implement linearly separable functions. It learns a decision boundary that separates two classes using a line (called a hyperplane) in the feature space.

Given

Two datasets are given. Dataset examples are of the form : $(x_{n1}, x_{n2}, ...x_{nD,}\ t_n)$
where

- x_n is a D dimensional unit vector for n ∈ {1,2,…...,N}
- t_n is the target attribute (In the the given examples t_n = 0 if $x_n \in$ C1 and t_n = 1 if $x_n \in$ C2) for n∈{1,2,…...,N}

Both the datasets are divided into 70:30 ratio as training and testing sets respectively.

Procedure

In this algorithm, the D input features are transformed to M mature features $\Phi(x)$.

$$\Phi_M (x_n) = \Phi_M(x_{n1,} x_{n2,} \ldots\ldots, x_{nD})$$

For the given datasets we assume that the features are already mature i.e we take the feature vector $\Phi(x)$ as x itself.

The feature vector $\Phi(x)$ is then used to construct a generalized linear model of the form

$$y(x) = f(w^T \Phi(x))$$

where f() is a step function.

The target attribute values are chosen as t =1 and t= -1.

To determine w , we use the perceptron criterion. It follows that,

$$w^T \Phi (x)\, t_n \geq 0 \quad \text{for correctly classified examples}$$
$$w^T \Phi (x)\, t_n < 0 \quad \text{for misclassified examples}$$

Therefore, for a misclassified example, the algorithm tries to minimize the quantity $-\, w^T \Phi (x)\, t_n$. The perceptron criterion is therefore given by

$$E_p = \sum_{n \in M} w^T \Phi (x)\, t_n$$

where M denotes the set of all misclassified patterns.

By applying the stochastic gradient descent algorithm, the change in the weight vector w is given by

$$w^{(\tau+1)} = w^{(\tau)} + \eta \Phi_n t_n$$

where η is the learning rate parameter and τ is an integer that indexes the steps of the algorithm.
We cycle through the training patterns in turn, and for each pattern x_n we evaluate the perceptron function. If the pattern is correctly classified, then the weight vector remains unchanged, whereas if it is incorrectly classified, then for class t =1 we add the vector $\Phi(x_n)$ onto the current estimate of weight vector w while for class t= -1 we subtract the vector $\Phi(x_n)$ from w. By performing 10^6 iterations on the training set, we get the parameter w.

3.2. Accuracy of your model on both the datasets

After performing 10^6 iterations on the training set, we get the following results.

```
Dataset1
Accuracy in training data =  0.9854318418314256
Accuracy in testing data 0.975669099756691
Not converged after 1000000 iterations


Dataset2
Accuracy in training data =  1.0
Accuracy in testing data 1.0
Converged after  47  iterations
```

3.3. Dataset which was more linearly separable

Since the perceptron algorithm converged much before for Dataset 2 (within 10^6 iterations) as compared to dataset 1 therefore Dataset 2 is more linearly separable.
Hence the obtained accuracy of Dataset 2 is greater than that of Dataset 1.

3.4. Major limitations of the Perceptron classifier.

Below are the limitations of perceptron algorithm:

1. The output values of a perceptron can take on only one of two values (e.g. -1 or 1). So this algorithm can only be used for binary classification.
2. The Perceptron algorithm can only classify linearly separable data.

3. For data sets that are not linearly separable, the perceptron learning algorithm will never converge.
4. There is no specific time limit within which the algorithm will provide the result.